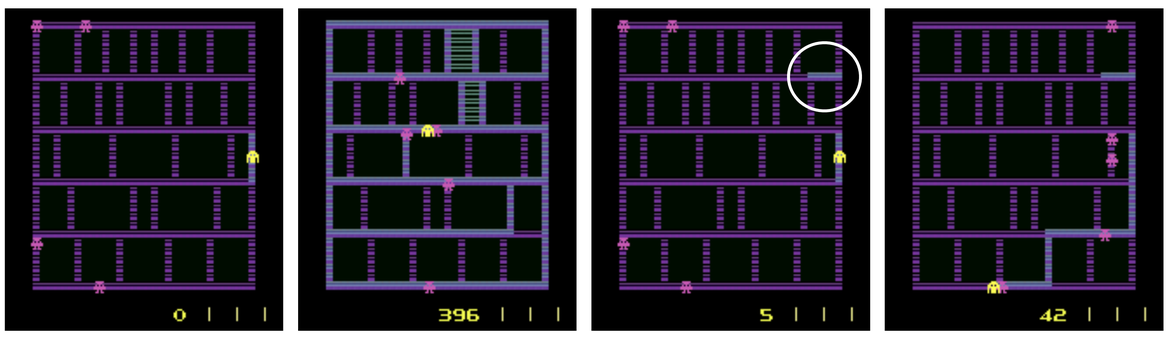
Generalization in Deep Reinforcement Learning

Sam Witty[1], Jun Ki Lee[2], Emma Tosch[1], Akanksha Atrey[1], Michael Littman[2], David Jensen[1]

[1]University of Massachusetts Amherst, [2]Brown University

1. Motivation

To what extent do the accomplishments of deep RL agents demonstrate generalization, and how can we recognize such a capability when presented with only a black-box controller?



Default start | Default death | Modified start | Modified death

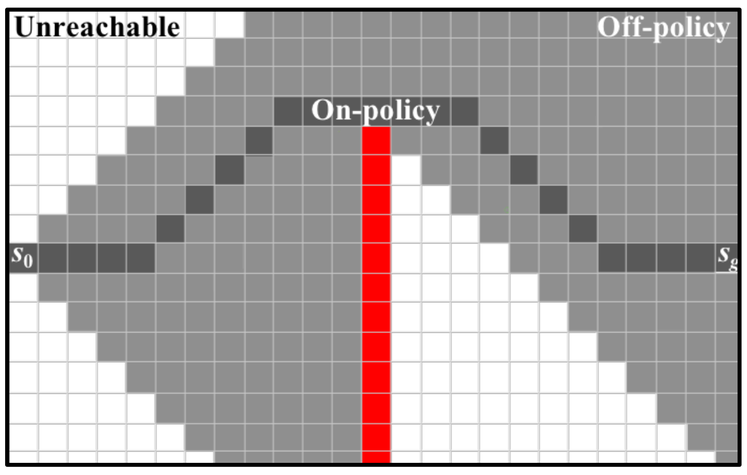
For example, an agent trained to play the Atari game of Amidar achieves large rewards when evaluated from the default initial state, but **small *non-adversarial* modifications dramatically degrade performance**.

2. Recasting Generalization

Naïve evaluation of a policy on held-out training states only measures an agent's ability to *use* data after it is collected. Using this method, we could incorrectly claim that an agent has generalized, even if it only performs well on a small subset of states.

We partition the universe of possible input states into three sets, according to how the agent can encounter them following its learned policy π from $s_0 \in S_0$, the set of initial states.

- **On-policy states**, S_{on}, can be encountered by following π from some s_0.
- **Off-policy states**, S_{off}, can be encountered by following any $\pi' \in \Pi$, the set of all policy functions.
- **Unreachable states**, $S_{\text{unreachable}}$, can not be encountered by following any $\pi' \in \Pi$, but are still in the domain of the state transition function $T(s, a, s')$.



In this grid-world example, the agent can take actions *up-right, right,* and *down-right*.

We define a q-value based agent's generalization abilities via the following, where δ and β are small positive values. $v^*(s)$ is the optimal state-value, $v_\pi(s)$ is the actual state-value by following π, and $\hat{v}(s)$ is the estimated state-value. $q^*(s,a)$, $q_\pi(s,a)$, and $\hat{q}(s,a)$, are the corresponding state-action values.

Definition 1 (Repetition) *An RL agent has high repetition performance, G_R, if $\delta > |\hat{v}(s) - v_\pi(s)|$ and $\beta > v^*(s) - v_\pi(s)$, $\forall s \in S_{on}$.*

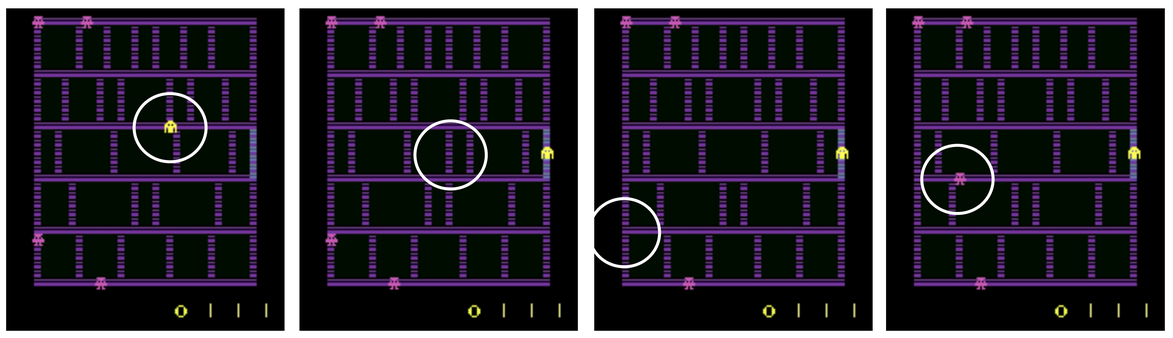
Definition 2 (Interpolation) *An RL agent has high interpolation performance, G_I, if $\delta > |\hat{q}(s,a) - q_\pi(s,a)|$ and $\beta > q^*(s,a) - q_\pi(s,a)$, $\forall s \in S_{off}, a \in A$.*

Definition 3 (Extrapolation) *An RL agent has high extrapolation generalization, G_E, if $\delta > |\hat{q}(s,a) - q_\pi(s,a)|$ and $\beta > q^*(s,a) - q_\pi(s,a)$, $\forall s \in S_{unreachable}, a \in A$.*

Why unreachable states? An agent interacting in the grid-world environment learns tabular q-values, therefore we should not expect it to satisfy any reasonable definition of generalization. Given enough exploration, $\hat{v}(s)$ would converge to $v^*(s)$ for all $s \in S_{\text{off}}$. **Only the definition G_E is consistent with this intuition**, that function-approximation is necessary to achieve generalization.

3. Empirical Methodology

Given a parameterized simulator we can intervene on individual components of latent state and forward-simulate an agent's trajectory through the environment.



Player random start (PRS) | Added line segment (ALS) | Enemy removal (ER) | Enemy shifted (ES)

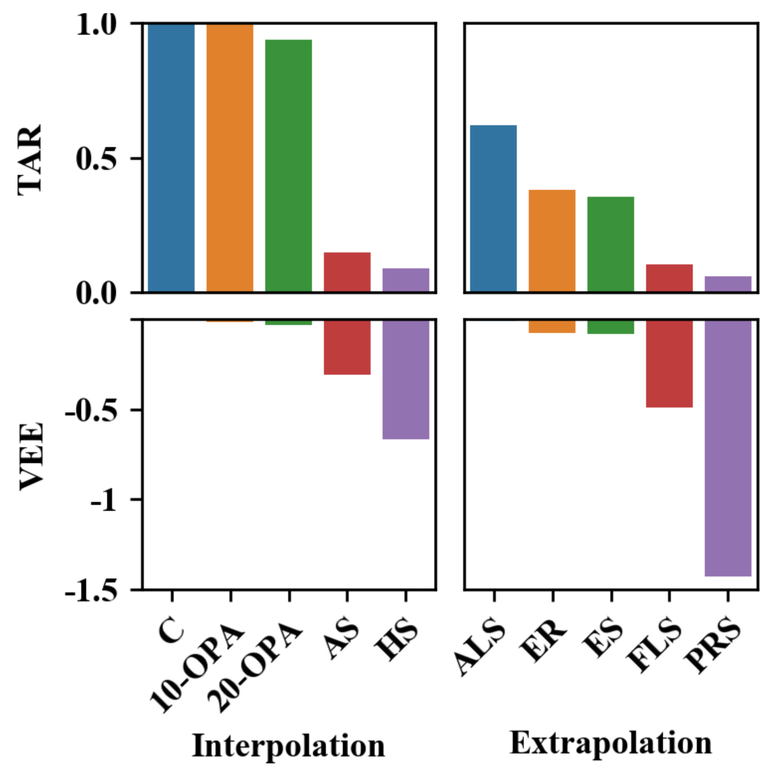
Intervening on individual components of latent state produces unreachable states, enabling empirical tests of an agent's generalization capabilities.

We can generate off-policy states by having the agent take k random actions during its trajectory (k-OPA) or extracting states from the trajectories of alternative agents (AS) and human players [1] (HS).

4. Analysis Case-Study

To demonstrate theses ideas we implement Intervenidar, a fully parameterized version of the Atari game of Amidar. Unlike previous work on adversarial attacks [1], **interventions in Intervenidar change the latent state itself, not only the agent's perception of state.**

We train the state-of-the-art dueling network architecture, double Q-loss function, and prioritized experience replay [3,4,5] using the standard pixel-based Atari MDP specification [2] with the default start position of the original Amidar game. After convergence, we expose the agent to off-policy and unreachable states.



The agent's total accumulated reward (TAR) and value estimate error (VEE) degrade dramatically when exposed to off-policy and unreachable states. Our controlled experiments demonstrate that the agent is particularly brittle with respect to changes in player position (PRS) and the filled/unfilled status of line segments (FLS). Agent swaps (AS) and human starts (HS) produce states that differ from training data in these respects.

5. Conclusions

- We propose a novel characterization of a black-box RL agent's generalization abilities based on performance from on-policy, off-policy, and unreachable states.
- We provide empirical methods for evaluating a RL agent's generalization abilities using intervenable parameterized simulators.
- We demonstrate these empirical methods using Intervenidar, a parameterized version of the Atari game of Amidar. We find that the state-of-the-art dueling DQN architecture fails to generalize to small changes in latent state.

Acknowledgments — Thanks to Kaleigh Clary, Reilly Grant, and the anonymous reviewers for thoughtful comments and contributions. This material is based upon work supported by the United States Air Force under Contract No. FA8750-17-C-0120. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the United States Air Force.

[1] Huang, S. Papernot, N., Goodfellow, I., Duan, Y., Abbeel, P. Adversarial attacks on neural network policies. ArXiv preprint ArXiv:1702.02284, 2017.
[2] Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A., Veness, J., Bellemare, M., Graves, A., Riedmiller, M., and Fidjeland, A., Ostrovski, G. Human-level control through deep reinforcement learning. Nature, 2015.
[3] Schaul, T., Quan, J., Antonoglou, I., Silver, D. Prioritized Experience Replay. International Conference on Learning Representations, 2016.
[4] Van Hasselt, H., Guez, A., Silver, D. Deep Reinforcement Learning with Double Q-Learning. Proceedings of the Association for the Advancement of Artificial Intelligence, 2016.
[5] Wang, Z., Schaul, T., Hessel, M., Van Hasselt, H., and Lanctot, M., and De Freitas, N. Dueling network architectures for deep reinforcement learning. International Conference on Machine Learning, 2016.